



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/676,227	09/30/2003	Gunter Schwarzbauer	06896/100M097-US1	9707

7278 7590 01/12/2006

DARBY & DARBY P.C.
P. O. BOX 5257
NEW YORK, NY 10150-5257

EXAMINER

RUTLEDGE, AMELIA L

ART UNIT	PAPER NUMBER
----------	--------------

2176

DATE MAILED: 01/12/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No. 10/676,227	Applicant(s) SCHWARZBAUER ET AL.	
	Examiner Amelia Rutledge	Art Unit 2176	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 30 September 2003.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-31 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-31 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 30 September 2003 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|---|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| 3) <input checked="" type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date <u>4/8/04</u> . | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

1. This action is responsive to communications: original application, filed 09/30/2003.
2. Claims 1-31 are pending in the case. Claims 1 and 28-31 are independent claims.

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

4. **Claims 1-31 are rejected under 35 U.S.C. 103(a) as being unpatentable over Quinlan et al. (hereinafter "Quinlan"), U.S. Patent No. 6,397,253, issued May 2002, in view of Sidles, U.S. Pub. No. 2002/0062342, published May 2002.**

Regarding independent claim1, Quinlan teaches a client system for accessing facilities of a server system through a communications network, where the client system includes a high performance gateway component that operates in conjunction with a standard browser component (Col. 5, l. 12-20), compare to *A system for automatic context management for testing, monitoring and automating network applications, comprising network applications with client side code execution, said system comprising: a testing, monitoring, or automating tool.* Quinlan teaches a contextual API

Art Unit: 2176

at the page level for a replay engine, i.e., the gateway component which records session information (Col. 13, l. 53-Col. 15, l. 18).

While Quinlan does not explicitly teach a recorder capable of recording at least one context-full test script, the replay engine being capable of executing said context-full test script, Sidles teaches an automatic form filling system using fuzzy logic where a history unit generates a new set of rules based on form context (p. 9, par. 88, 90), i.e., *recording at least one context-full test script* for filling the form. Sidles also teaches that a replay engine is capable of executing the test script. Both Quinlan and Sidles are analogous art, being directed toward the management of context for web applications. It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Sidles form filling system with test scripts to Quinlan's contextual session management, so that Quinlan would have the benefit of an automatic method of filling out forms which would speed up purchasing transactions (Sidles, p. 2, par. 16).

Regarding dependent claim 2, Quinlan teaches a contextual API at the page level for a replay engine, i.e., the gateway component which records session information (Col. 13, l. 53-Col. 15, l. 18).

Regarding dependent claims 3 and 4, Quinlan teaches that HTML documents are parsed. While Quinlan does not explicitly teach an extensible document parser for determining at least one parser extension, Sidles teaches a collection of data flow monitors customized to the characteristics and protocols of the operating systems within which they are embedded, and may be partially resident on a device and on a server (p. 5, par. 51). It would have been obvious to one of ordinary skill in the art at the time of

Art Unit: 2176

the invention that the data flow monitor used in conjunction with the form fill system (p. 5-6, par. 55-57) taught by Sidles is equivalent in function and description to the claimed extensible document parser. Sidles also teaches that a simple priority scheme teaches the appropriate data flow monitor to be selected, compare to a *replay instruction specifying at least one parser addition and parameters for said parser addition*. Both Quinlan and Sidles are analogous art, being directed toward the management of context for web applications. It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Sidles form filling system with test scripts to Quinlan's contextual session management, so that Quinlan would have the benefit of an automatic method of filling out forms which would speed up purchasing transactions (Sidles, p. 2, par. 16).

Regarding dependent claims 5 and 6, while Quinlan does not explicitly teach that the parser addition includes a plug-in module for said extensible document parser, Sidles teaches that the data flow monitor is installed as an "add on", i.e., plug-in module (p. 5, par. 52). Sidles also teaches that both the data flow monitor and the form filling system can be extended (p. 5, par. 51-52; p. 6, p. 6, par. 61). Both Quinlan and Sidles are analogous art, being directed toward the management of context for web applications. It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Sidles form filling system with test scripts to Quinlan's contextual session management, so that Quinlan would have the benefit of an automatic method of filling out forms which would speed up purchasing transactions (Sidles, p. 2, par. 16).

Regarding dependent claims 7-10, Quinlan teaches an API library and a parser (Col. 15, l. 1-5). While Quinlan does not explicitly teach a library of parser additions wherein each parser addition implements a specific parsing algorithm, Sidles teaches a dictionary database which is a component of the form filling system, which is loaded with form information and may be altered to add or delete information (p. 6, par. 61). The dictionary database and form fill system have rules for parsing and filling in forms (p. 8, par. 76, 79, 80), compare each rule of the databases to a *library of parser additions wherein each parser addition implements a specific parsing algorithm*. Both Quinlan and Sidles are analogous art, being directed toward the management of context for web applications. It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Sidles form filling system with test scripts to Quinlan's contextual session management, so that Quinlan would have the benefit of an automatic method of filling out forms which would speed up purchasing transactions (Sidles, p. 2, par. 16).

Regarding dependent claims 9 and 10, Quinlan teaches an algorithm for parsing forms from a HTML document (Col. 45-46, l. 1-30). Quinlan teaches an algorithm for parsing hyperlinks from an HTML document (Col. 12, l. 66-Col. 15, l. 58).

Regarding dependent claim 11, while Quinlan in view of Sidles does not explicitly teach an algorithm for parsing embedded documents from a HTML document, it was notoriously well known in the art at the time of the invention that HTML browsers had functionality for parsing HTML documents in order to display them to the user, including algorithms for parsing embedded documents.

Regarding dependent claim 12, Quinlan teaches an algorithm for parsing hyperlinks by searching between a left and right boundary string (Col. 12, l. 66-Col. 14, l. 41), using a protocol for URL fields.

Regarding dependent claim 13, Quinlan teaches a remote forms protocol which comprises form merging replay instructions (Col. 45-46, l. 29-30).

Regarding dependent claim 14, while Quinlan teaches a remote forms protocol which comprises form merging instructions (Col. 45-46, l. 25-30), Quinlan does not explicitly teach a reference to a form in the test script. However, Sidles teaches a reference to a previously downloaded form in a test script, i.e., to see if the user made changes to a field, and instructions for merging the form and scripts containing previously submitted form information to automatically fill out a form to be submitted (p. 12, par 120-123). Both Quinlan and Sidles are analogous art, being directed toward the management of context for web applications. It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Sidles form filling system with test scripts to Quinlan's contextual session management, so that Quinlan would have the benefit of an automatic method of filling out forms which would speed up purchasing transactions (Sidles, p. 2, par. 16).

Regarding dependent claim 15, while Quinlan does not explicitly teach instructions for merging each individual form field, Sidles teaches merging instructions for each individual form field of the HTML form and script form (p. 10-11, par. 102-104). Sidles teaches instructions for matching each field of the form to a dictionary entry for the form, and if a match cannot be found, the field information is not sent, compare to

an instruction to send a form field value obtained from said HTML form; an instruction to send a form field value specified in said script form; and an instruction to not send one of said form fields. Both Quinlan and Sidles are analogous art, being directed toward the management of context for web applications. It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Sidles form filling system with test scripts to Quinlan's contextual session management, so that Quinlan would have the benefit of an automatic method of filling out forms which would speed up purchasing transactions (Sidles, p. 2, par. 16).

Regarding dependent claim 16, Quinlan teaches a remote forms mechanism (Col. 6, l. 62-Col. 7, l. 37) with a unique URL coding all pertinent connection information and any available options into the URL (Col. 7, l. 26-36), compare to *an action URL in said test script instead of an action URL obtained from said HTML form for said form to be submitted.*

Regarding dependent claim 17, Quinlan teaches a contextual API at the page level for a replay engine, i.e., the gateway component which records session information (Col. 13, l. 53-Col. 15, l. 18). While Quinlan does not explicitly teach that the script comprises parser extensions and form merging instructions, Sidles teaches that both the data flow monitor and the form filling system can be extended (p. 5, par. 51-52; p. 6, p. 6, par. 61). Both Quinlan and Sidles are analogous art, being directed toward the management of context for web applications. It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Sidles form filling system with test scripts to Quinlan's contextual session management, so that Quinlan

would have the benefit of an automatic method of filling out forms which would speed up purchasing transactions (Sidles, p. 2, par. 16).

Regarding dependent claim 18, Quinlan teaches that the browser and gateway components record session history by recording the session information as an entry in a persistent session table (Col. 5, l. 40-56), compare to *wherein said recorder keeps track of a session history by building representations of all web pages downloaded so far during a recording session*.

Regarding dependent claim 19, Quinlan teaches a contextual API at the page level for a replay engine, i.e., the gateway component which records session information (Col. 13, l. 53-Col. 15, l. 18). Quinlan teaches that HTML documents are parsed.

Regarding dependent claim 20, while Quinlan does not explicitly teach that the gateway component automatically detects which parser extensions and form merging instructions are needed in order to record a test script which will correctly use dynamic information during a script replay, Sidles teaches an automatic method of form filling where the system tests which rules and instructions are needed in order to correctly fill a dynamic form (p. 12, par 125). Both Quinlan and Sidles are analogous art, being directed toward the management of context for web applications. It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Sidles form filling system with test scripts to Quinlan's contextual session management, so that Quinlan would have the benefit of an automatic method of filling out forms which would speed up purchasing transactions (Sidles, p. 2, par. 16).

Regarding dependent claim 21, while Quinlan does not explicitly teach that the recorder detects the need for recording at least one of said parser extensions by detecting that a context-less replay instruction is to be recorded otherwise, Sidles teaches a process performed by the completed form analysis engine for detecting the need for recording parser extensions, i.e., dictionary or database entries (p. 11, par. 107-113) and detecting whether instructions should be recorded with or without context (p. 11, par. 110-112). Both Quinlan and Sidles are analogous art, being directed toward the management of context for web applications. It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Sidles form filling system with test scripts to Quinlan's contextual session management, so that Quinlan would have the benefit of an automatic method of filling out forms which would speed up purchasing transactions (Sidles, p. 2, par. 16).

Regarding dependent claim 22, claim 22 is directed toward substantially similar subject matter as claimed in dependent claims 3, 4, 7, and 8 and is rejected along the same rationale, except where Claim 22 recites the additional limitation: *said recorder detects which one of said parser extensions is to be recorded by querying each of said parser additions for suitable parameters*. While Quinlan does not explicitly teach the claimed limitation, Sidles teaches recording new rules for forms by querying a history database for suitable parameters (p. 7, par. 88). Both Quinlan and Sidles are analogous art, being directed toward the management of context for web applications. It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Sidles form filling system with test scripts to Quinlan's contextual session

management, so that Quinlan would have the benefit of an automatic method of filling out forms which would speed up purchasing transactions (Sidles, p. 2, par. 16).

Regarding dependent claims 23 and 24, while Quinlan does not explicitly teach fuzzy form detection, Sidles teaches a method of fuzzy form detection comparing a form being submitted to all forms in a history database (p. 9, par. 86-90), choosing a form from said session history which is most similar to said form being submitted. Sidles teaches generating logic rules for the form to be used to complete the form fields (p. 9, par. 90), compare to *recording said form merging instructions so that said recorded form merging instructions applied to said form chosen from said session history result in a form identical to said form being submitted*. That is, the dictionary database is adjusted so that the automatic filler program and the rules engine will fill out that form the next time it is encountered (p. 10, par. 91, l. 1-8). Both Quinlan and Sidles are analogous art, being directed toward the management of context for web applications. It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Sidles form filling system with test scripts to Quinlan's contextual session management, so that Quinlan would have the benefit of an automatic method of filling out forms which would speed up purchasing transactions (Sidles, p. 2, par. 16).

Regarding dependent claim 25, while Quinlan does not explicitly teach that the replay engine executes a test script with parser extensions and form merging instruction, Sidles teaches an automatic form filling system using fuzzy logic where a history unit generates a new set of rules based on form context (p. 9, par. 88, 90), i.e., *recording at least one context-full test script* for filling the form. Sidles also teaches that

Art Unit: 2176

a replay engine is capable of executing the test script. Both Quinlan and Sidles are analogous art, being directed toward the management of context for web applications. It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Sidles form filling system with test scripts to Quinlan's contextual session management, so that Quinlan would have the benefit of an automatic method of filling out forms which would speed up purchasing transactions (Sidles, p. 2, par. 16).

Regarding dependent claim 26, Quinlan teaches that the browser and gateway components record session history by recording the session information as an entry in a persistent session table (Col. 5, l. 40-56), compare to *wherein said replay engine keeps track of a session history by building representations of all web pages downloaded so far during a replaying session*.

Regarding dependent claim 27, Quinlan teaches a contextual API at the page level for a replay engine, i.e., the gateway component which records session information (Col. 13, l. 53-Col. 15, l. 18). Quinlan teaches that HTML documents are parsed.

Regarding independent claim 28, Quinlan teaches a method of comparing a form being submitted to forms in a session history (Col. 6, l. 62-Col. 7, l. 16). While Quinlan does not explicitly teach recording form merging instructions for fuzzy form detection, Sidles teaches a method of fuzzy form detection where a form is chosen from a database of forms which is similar to the form being submitted, and merged with past form data to produce a new filled form (p. 6, par. 16-62; p. 9, par. 86-90), compare to claim 28, *choosing a form from said session history which is similar to said form being*

Art Unit: 2176

submitted; recording form merging instructions to said form chosen from said session history so that a resulting form is substantially identical to said form being submitted.

Both Quinlan and Sidles are analogous art, being directed toward the management of context for web applications. It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Sidles form filling system with test scripts to Quinlan's contextual session management, so that Quinlan would have the benefit of an automatic method of filling out forms which would speed up purchasing transactions (Sidles, p. 2, par. 16).

Regarding independent claim 29, Quinlan teaches a device for automatic context management for testing, monitoring and automating network applications, comprising network applications with client side code execution (Col. 5, l. 26-56). Quinlan teaches a processor; and a memory storing processing instructions for controlling the processor (Col. 7, l. 65-Col. 8, l. 26). While Quinlan does not explicitly teach instructions to record at least one context-full test script, the replay engine being capable of executing said context-full test script, Sidles teaches an automatic form filling system using fuzzy logic where a history unit generates a new set of rules based on form context (p. 9, par. 88, 90), i.e., *record at least one context-full test script* for filling the form. Sidles also teaches that a replay engine is capable of executing the test script. Both Quinlan and Sidles are analogous art, being directed toward the management of context for web applications. It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Sidles form filling system with test scripts to Quinlan's contextual session management, so that Quinlan would have

the benefit of an automatic method of filling out forms which would speed up purchasing transactions (Sidles, p. 2, par. 16).

Regarding independent claim 30, Quinlan teaches a device for automatic context management for testing, monitoring and automating network applications, comprising network applications with client side code execution (Col. 5, l. 26-56). Quinlan teaches a processor; and a memory storing processing instructions for controlling the processor (Col. 7, l. 65-Col. 8, l. 26). Quinlan teaches an algorithm for parsing hyperlinks by searching between a left and right boundary string (Col. 12, l. 66-Col. 14, l. 41), using a protocol for URL fields. Quinlan teaches a contextual API at the page level for a replay engine, i.e., the gateway component which records session information (Col. 13, l. 53-Col. 15, l. 18), to execute scripts.

While Quinlan does not explicitly teach instructions to record at least one context-full test script, the replay engine being capable of executing said context-full test script, Sidles teaches an automatic form filling system using fuzzy logic where a history unit generates a new set of rules based on form context (p. 9, par. 88, 90), i.e., *record at least one context-full test script* for filling the form. Sidles also teaches that a replay engine is capable of executing the test script.

Sidles teaches an automatic method of form filling where the system tests which rules and instructions are needed in order to correctly fill a dynamic form (p. 12, par 125). Sidles teaches a dictionary database which is a component of the form filling system, which is loaded with form information and may be altered to add or delete information (p. 6, par. 61). The dictionary database and form fill system have rules for

parsing and filling in forms (p. 8, par. 76, 79, 80), compare recording and applying each rule of the databases to claim 30; *select said parser addition for said parser extension from a library of parser additions; implement a specific parsing algorithm using each of said parser additions of said library of parser additions.*

Both Quinlan and Sidles are analogous art, being directed toward the management of context for web applications. It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Sidles form filling system with test scripts to Quinlan's contextual session management, so that Quinlan would have the benefit of an automatic method of filling out forms which would speed up purchasing transactions (Sidles, p. 2, par. 16).

Regarding independent claim 31, Quinlan teaches a device for automatic context management for testing, monitoring and automating network applications, comprising network applications with client side code execution (Col. 5, l. 26-56). Quinlan teaches a processor; and a memory storing processing instructions for controlling the processor (Col. 7, l. 65-Col. 8, l. 26). Quinlan teaches a contextual API at the page level for a replay engine, i.e., the gateway component which records session information (Col. 13, l. 53-Col. 15, l. 18), to execute scripts.

While Quinlan does not explicitly teach a recorder capable of recording at least one context-full test script, the replay engine being capable of executing said context-full test script, Sidles teaches an automatic form filling system using fuzzy logic where a history unit generates a new set of rules based on form context (p. 9, par. 88, 90), i.e., *record at least one context-full test script* for filling the form. Sidles also teaches that a

Art Unit: 2176

replay engine is capable of executing the test script. Sidles teaches a reference to a form in a previously downloaded web page and a reference to rules to complete the form and validate the fields, and instructions for merging said HTML form and said script form to produce a form to be submitted (p. 9, par. 88-90).

Both Quinlan and Sidles are analogous art, being directed toward the management of context for web applications. It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Sidles form filling system with test scripts to Quinlan's contextual session management, so that Quinlan would have the benefit of an automatic method of filling out forms which would speed up purchasing transactions (Sidles, p. 2, par. 16).

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Amelia Rutledge whose telephone number is 571-272-7508. The examiner can normally be reached on Monday - Friday 9:30 - 6:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Heather Herndon can be reached on 571-272-4136. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Art Unit: 2176

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

AR


HEATHER R. HERNDON
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100